General Disclaimer

One or more of the Following Statements may affect this Document

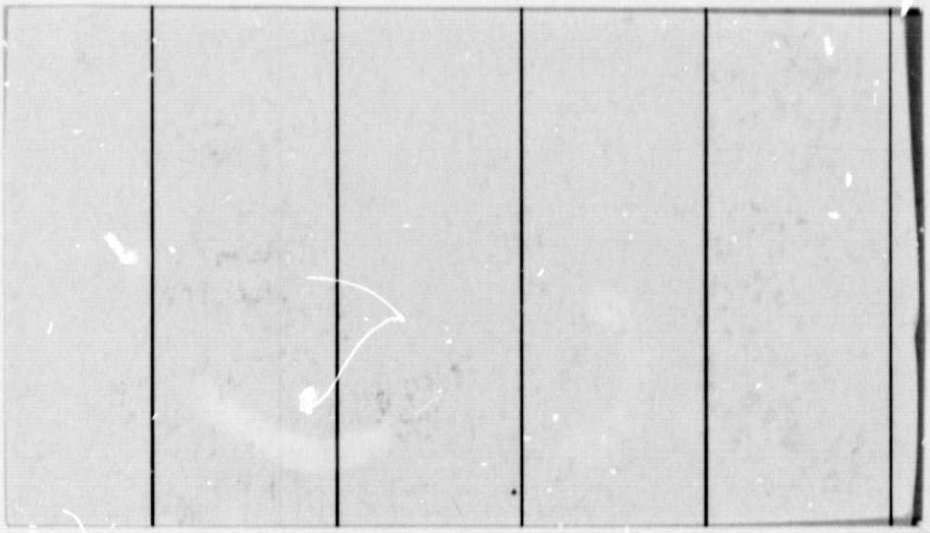
- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.

- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.

- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.

- This document is paginated as submitted by the original source.

- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.



Department of

ELECTRICAL ENGINEERING

UNIVERSITY OF NOTRE DAME, NOTRE DAME, INDIANA

Short, Unit-Memory, Byte-Oriented, Binary Convolutional Codes Having Maximal Free Distance*

Technical Report No. EE-754

Lin-nan Lee
Department of Electrical Engineering
University of Notre Dame
Notre Dame, Indiana 46556

July 11, 1975

Abstract: It is shown that (n_o, k_o) convolutional codes with unit memory always achieve the largest free distance among all codes of the same rate k_o/n_o and same number 2^{Mk_o} of encoder states, where M is the encoder memory. A unit-memory code with maximal free distance is given at each place where this free distance exceeds that of the best code with k_o and n_o relatively prime, for all $Mk_o \leq 6$ and for R = 1/2, 1/3, 1/4, 2/3. It is shown that the unit-memory codes are byte-oriented in such a way as to be attractive for use in concatenated coding systems.

* This research was supported by the National Aeronautics and Space Administration under NASA Grant NSG 5025 at the University of Notre Dame in liaison with the Goddard Space Flight Center.

I. INTRODUCTION

Let the binary k_o-tuple $\underline{a}_t$ denote the subblock of information digits at time t (t = 0, 1,2,...), and let the binary n_o-tuple $\underline{b}_t$ denote the encoded subblock at time t in an (n_o, k_o) convolutional code. Then, the encoding equations may be written

$$\underline{b}_t = \underline{a}_t G_0 + \underline{a}_{t-1} G_1 + \ldots + \underline{a}_{t-M} G_M \quad (t = 0, 1,2,\ldots) \qquad (1)$$

where each G_i is a $k_o \times n_o$ binary matrix, where M is the code memory, where the operations are in GF(2), and where, by way of convention, $\underline{a}_t = \underline{0}$ for $t < 0$. An encoding circuit is shown in Figure 1. Note that the encoder has 2^{Mk_o} distinct states where the state is taken as the contents of the delay cells in the encoder. We shall refer to the number Mk_o of binary state variables in the encoder as the state-complexity of the convolutional code.

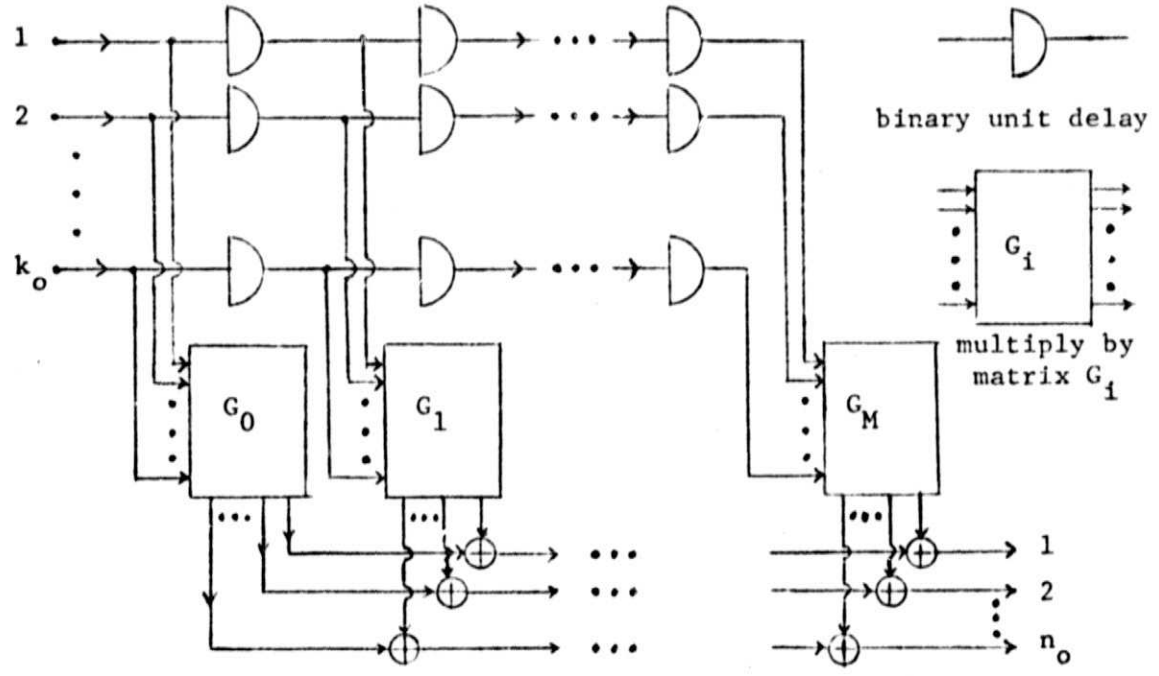


Fig. 1: An encoding circuit for an (n_o, k_o) convolutional code with memory M.

The constraint length K (measured in information digits) of the convolutional code is defined by

$$K = (M + 1)k_c$$

The rate R of the code is defined by

$$R = k_o/n_o.$$

In virtually all past applications of convolutional codes, k_o and n_o have been taken as relatively prime, i.e., gcd $(k_o, n_o) = 1$ where "gcd" denotes "greatest common divisor." In fact, the condition gcd $(k_o, n_o) = 1$ is generally tacitly assumed so that speaking, for instance, of a convolutional code as being of rate R = 1/2 would imply $k_o = 1$ and $n_o = 2$ unless the contrary were explicitly stated. As will be seen, however, there can be advantages in taking gcd $(k_o, n_o) > 1$.

For convenience, let $\underline{b}_{[t,t']}$ denote the encoded sequence $[\underline{b}_t : \underline{b}_{t+1} : \ldots : \underline{b}_{t'}]$ over time units t through t' and let $\underline{b}_{[0,\infty)}$ denote the entire semi-infinite encoded sequence. Let $\underline{a}_{[t,t']}$ and $\underline{a}_{[0,\infty)}$ be similarly defined. The free distance, d_{free}, of the convolutional code is the minimum Hamming distance between all pairs of encoded sequences $\underline{b}_{[0,\infty)}$ resulting from pairs of information sequences $\underline{a}_{[0,\infty)}$ that differ in their time 0 subblock. By the linearity of (1), it follows that

$$d_{free} = \min_{\underline{a} \neq 0} W(\underline{b}_{[0,\infty)}) \qquad (2)$$

where $W(\cdot)$ denotes the Hamming weight of the enclosed binary sequence and where the minimization is over all $\underline{a}_{[0,\infty)}$ such that $\underline{a}_0 \neq \underline{0}$. The code is non-catastrophic [1,2] when $W(\underline{a}_{[0,\infty)}) = \infty$ implies that $W(\underline{b}_{[0,\infty)}) = \infty$, in which case the minimization in (2) reduces to the minimum over all $\underline{a}_{[0,\infty)}$ such that $\underline{a}_0 \neq 0$ and $W(\underline{a}_{[0,\infty)}) < \infty$. The restriction to non-catastrophic codes entails no loss in the achievable value

of d_{free} for given n_o, k_o and M, a fortunate situation because the non-catastrophic property is essential in applications (cf. [1]).

Because d_{free} is the primary determiner of decoding error probability when Viterbi [i.e., maximum likelihood] decoding is used with a non-catastrophic code, d_{free} is the usual criterion of goodness for codes to be used with Viterbi decoders. Because the number of states of the Viterbi decoder [5] coincides with the number of encoder states, viz. 2^{Mk_o}, practicality dictates a small state-complexity. The region $Mk_o \leq 6$ appears to be about the range where Viterbi decoding is attractive in applications. Thus, for Viterbi decoding applications, we are motivated to find, for a given code rate and a given state-complexity in the above range, a convolutional code with maximum d_{free}. In the next section, we report the results of our search for such codes and we also derive a useful upper bound on the attainable d_{free}.

II. UNIT-MEMORY CODES WITH MAXIMAL d_{free}

Any (n_o, k_o) convolutional code with memory M can be considered as an $(n_o' = Mn_o, k_o' = Mk_o)$ code with $M' = 1$ simply by taking

$$G_0' = \begin{bmatrix} G_0 & G_1 & \cdots & G_{M-1} \\ 0 & G_o & \cdots & G_{M-2} \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & G_0 \end{bmatrix}$$

and

$$G_1' = \begin{bmatrix} G_M & 0 & \cdots & 0 \\ G_{M-1} & G_M & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ G_1 & G_2 & \cdots & G_M \end{bmatrix}.$$

These two codes are entirely equivalent in the sense that the same semi-infinite binary information sequence produces the same semi-infinite binary encoded sequence, although the division into time subblocks would be different. Since

the state-complexity is Mk_o for both codes, it follows that <u>for a given state-complexity and given rate, the maximum value of d_{free} is achieved within the subset of convolutional codes with</u> $M = 1$. Hence, we can restrict our search for optimal codes, for a given rate and state-complexity, to codes with <u>unit-memory</u>.

For a unit-memory code, equation (1) reduces to

$$\underline{b}_t = \underline{a}_t G_0 + \underline{a}_{t-1} G_1 \quad (t = 0, 1, 2, \ldots). \tag{3}$$

When $\underline{a}_0 \neq \underline{0}$ is the only non-zero information subblock, then

$$\underline{b}_{[0,1]} = \underline{a}_0 [G_0 : G_1] \tag{4}$$

is the only possibly-non-zero protion of $\underline{b}_{[0,\infty)}$. From (2), it then follows that <u>the attainable d_{free} of a unit-memory (n_o, k_o) convolutional code is upper-bounded by the largest minimum distance of an $(n = 2n_o, k = k_o)$ block code</u>. We shall call this upper bound the <u>block code upper bound on d_{free}</u>, and we note that McEliece and Rumsey [6] have used similar arguments to derive more elaborate upper bounds on d_{free} for codes where $M \neq 1$.

We see that the argument used above to establish the block code upper bound on d_{free} suggests the following search procedure for finding a non-catastrophic unit-memory (n_o, k_o) code with maximal d_{free}:

(i) Set d equal to the largest d_{min} achievable by any $(n = 2n_o, k = k_o)$ block code.

(ii) Choose $[G_0 : G_1]$ as the generator matrix of a $(n = 2n_o, k = k_o)$ block code with $d_{min} = d$. If $d_{free} = d$ and the code is non-catastrophic, stop. Otherwise continue with step (ii) until all block codes with $d_{min} = d$ have been exhausted.

(iii) Reduce d by 1 and return to step (ii).

The above search procedure was carried out to obtain, for rates 1/4, 1/3, 1/2 and 2/3 (which are the usual rates of interest in applications), a

non-catastrophic unit-memory convolutional code with maximal d_{free} for all state-complexities of 6 or less. The values of d_{free} obtained are given in Table I where we also list, for comparison, the largest d_{free} attainable by a code of the same state-complexity having gcd $(n_o, k_o) = 1$. The block code upper bound on d_{free} for each case is also listed.

The codes with gcd $(n_o, k_o) = 1$ that achieve the values of d_{free} given in Table I may be found in Larsen [3] and Paaske [4]. The values of the block code upper bound on d_{free}, given in Table I, were taken from Calabi and Myrvaagnes [7] and from Helgert and Stinoff [8]. In Table II, we give the matrices G_0 and G_1 of a non-catastrophic unit-memory code with maximal d_{free} at each place where Table I shows that value to exceed the d_{free} available from the best gcd $(n_o, k_o) = 1$ code having the same state-complexity.

Rate (k_o/n_o)	State Complexity (No. of State Variables)	Block Code Upper Bound on d_{free}	Maximal d_{free} of Unit-Memory Codes	Maximal d_{free} of gcd $(n_o, k_o) = 1$ Codes
1/4	1	8	7	7
	2	10	10	10
	3	13	13	13
	4	16	16	16
	5	20	20	18
	6	24	24	20
1/3	1	6	5	5
	2	8	8	8
	3	10	10	10
	4	12	12	12
	5	15	15	13
	6	16	16	15
1/2	1	4	3	3
	2	5	5	5
	3	6	6	6
	4	8	8	7
	5	9	9	8
	6	10	10	10
2 2/3	2	4	3	3
	4	6	6	5
	6	8	7	7

Table I. Maximal d_{free} for a given state-complexity Mk_o of unit memory convolutional codes and of convolutional codes with gcd $(n_o, k_o) = 1$.

Rate	n_o	k_o	G_1	G_2	d_{free}
1/2	8	4	10000111 01001011 00101101 00011110	10001011 11100010 10111000 11010001	8
1/2	10	5	1000011111 0100001111 0010011110 0001011001 0000110101	1111111111 1111000000 0010110100 1010011010 0110101001	10
1/3	15	5	100001111011010 010000111101101 001001011110110 000101101101011 000011110110101	000111110010100 001101100101010 011001001100101 110000011110010 100010111001001	15
1/3	18	6	111000110100110000 011100011010011000 001110001101001100 000111100110000110 100011010011000011 110001101001100001	000011000111001011 000110001110010110 001100011100101100 011000111000011001 110000110001110010 100001100011100101	16
1/4	20	5	10000011111110011000 01000101110111001100 00100110110011100110 00010111011001100011 00001111101100110001	00011110100001100111 00110011010110001110 01100101101000111100 11000010110011011001 10001101011100010011	20
1/4	24	6	100000111110111010110000 010000011111011101011000 001000101111101110001100 000100110111010111000110 000010111011101011000011 000001111101110101100001	001111000101001011010011 011110001010010110100110 111100010100101100001101 111001101000011001011010 110011010001110010110100 100111100010100101101001	24
2/3	6	4	100001 010010 001011 000111	111100 110110 010011 101001	6

Table 2. The encoding matrices of some non-catastrophic unit-memory convolutional codes with free distance greater than the maximal free distance of gcd (n_o, k_o) = 1 codes of the same rate and state complexity.

III. BYTE-ORIENTED NATURE OF UNIT-MEMORY CODES

We now show that short, unit-memory convolutional codes are "byte-oriented" in such a way as to be attractive for use, with Viterbi decoding, as the inner coding component of a concatenated coding system.

In general, the state at time t of the convolutional encoder is the information sequence $a_{[t-M,t-1]} = [\underline{a}_{t-M}:\underline{a}_{t-M+1}: \cdots :\underline{a}_{t-1}]$ over the preceding M time units. Note that the successor of this state, namely, $[\underline{a}_{t-M+1}: \cdots :\underline{a}_{t-1}:\underline{a}_t]$, is already determined up to the 2^{k_o} choices of $\underline{a}_t$, i.e., each state will have 2^{k_o} successors in the "trellis" defined by the convolutional code [5]. In the corresponding Viterbi decoder, the "metric" for the best path to each of the 2^{Mk_o} possible states at time t must be relayed to each of its 2^{k_o} successors. Hence, the value of k_o influences the overall complexity of the Viterbi decoder, although much less strongly then does the state-complexity. Nonetheless, to determine, for instance, whether the Viterbi decoder for an R = 1/3, M = 1, k_o = 6 code (state-complexity 6) is simpler than the Viterbi decoder for an R = 1/3, M = 7, k_o = 1 code (state-complexity 7) would require a detailed analysis of the specific decoder design.

With Viterbi decoding of convolutional codes, there is a natural segmentation of the decoded information digits into bytes of k_o bits, because the information byte $\underline{a}_t$ acts as a unit in determining the correct state. The typical decoding "error events" [2] give rise to the incorrect decoding of a small number of bytes, viz., the average number of non-zero information bytes in an information sequence $\underline{a}_{[0,\infty)}$ that generates an encoded sequence $\underline{b}_{[0,\infty)}$ of Hamming weight d_{free}. In an M = 1 code, this average is near 1, although there will generally be about $k_o/2$ bit errors per byte. In a code with k_o = 1, there will generally be a small number (say, about 3) of byte errors but, since a byte is a bit in this case, the same number of bit errors. However, if the information bits

with the latter code are gathered into "bytes" of $Mk_o = M$ bits, there will generally be about $1\frac{1}{2}$ "byte" errors per error event since the bit decoding errors are not synchronized to begin at the start of these "bytes." Thus, even if both codes have the same state-complexity and same d_{free}, one would expect the unit-memory code to have a lower byte-error probability for bytes of Mk_o bits.

To test the validity of these observations, we simulated Viterbi decoding on an additive white Gaussian noise (AWGN) channel with several values of the energy per information bit to one-sided noise power spectral density ratio, E_b/N_o, for several convolutional codes of rate $R = 1/3$. The codes tested were (i) the $k_o = 6$ unit-memory code of Table II, (ii) the $k_o = 1$, $M = 6$ code given by Larsen [3], and (iii) the $k_o = 1$, $M = 7$ code given by Larsen [3]. The byte size was 6 bits. The results of the simulation are given in Table III. The unit-memory code (i) had a decoding byte-error probability about one-half that of the gcd $(n_o, k_o) = 1$ code (ii) with the same state-complexity ($Mk_o = 6$). This superiority of code (i) over code (ii) is due mainly to its greater free distance, 16 as opposed to 15. But we also see from Table III that the decoding byte-error probability of code (i) is about two-thirds that of code (iii) which has the same free distance 16 and greater state-complexity, 7 versus 6. This superiority of code (i) over code (iii) is due entirely to its byte-oriented nature.

We conclude that the unit-memory codes, because of their byte-oriented nature, appear attractive for use as the inner code in concatenated coding systems [9] where the outer code is a Reed-Solomon (RS) code over the alphabet $GF(2^{k_o})$, i.e. the bytes of the convolutional code are single digits for the RS code. For instance, code (i) above would be used with an RS code over $GF(2^6)$. We expect in the future to report in detail on the effectiveness of unit-memory codes in such applications.

E_b/N_o	M=1, (18,6) code		M=6, (3,1) code		M=7 (3,1) code	
(dB)	Byte-Error Prob.	95% Confidence	Byte-Error Prob.	95% Confidence	Byte-Error Prob.	95% Confidence
1.00	0.02950	±0.00533	0.04375	±0.00681	0.04000	±0.00619
1.25	0.01925	±0.00435	0.03250	±0.00561	0.02250	±0.00469
1.50	0.01100	±0.00329	0.02325	±0.00477	0.01400	±0.00372
1.75	0.00625	±0.00250	0.01275	±0.00350	0.01025	±0.00319

Table III. Byte-error probability for Viterbi decoding of three R = 1/3 convolutional codes on a simulated AWGN channel.

ACKNOWLEDGEMENT

The author would like to thank Professor James L. Massey who not only suggested to the author the possibility of constructing unit-memory convolutional codes superior to gcd $(n_o, k_o) = 1$ codes of the same state-complexity, but also supervised this work during the course of this research.

REFERENCES

[1] J. L. Massey and M. K. Sain, "Inverses of Linear Sequential Circuits," IEEE Transactions on Computers, Vol. C-17, pp. 330-337, April 1968.

[2] G. D. Forney, Jr., "Convolutional Codes I: Algebraic Structure," IEEE Transactions on Information Theory, Vol. IT-16, pp. 720-738, November 1970.

[3] Larsen, K. J., "Short Convolutional Codes with Maximal Free Distance for Rates 1/2, 1/3, and 1/4," IEEE Transactions on Information Theory, Vol. IT-19, pp. 371-372, May 1973.

[4] Paaske, E., "Short Binary Convolutional Codes with Maximal Free Distance for Rates 2/3, and 3/4," IEEE Transactions on Information Theory, Vol. IT-20. pp. 683-688, Sept. 1974.

[5] A. J. Viterbi, "Convolutional Codes and Their Performance in Communication Systems," IEEE Transactions on Communications, Vol. COM-19, pp. 751-772, October 1971.

[6] McEliece, R. J. and Rumsey, H. C., "Capabilities of Convolutional Codes," Jet Prop. Lab., Calif. Inst. of Tech., Space Programs Summary 37-49, Vol. 3, pp. 248-251, April 1968.

[7] Calabi, L. and Myrvaagnes, E., "On the Minimal Weight of Binary Group Codes," IEEE Transactions on Information Theory, Vol. IT-10, pp. 385-387, October 1964.

[8] Helgert, H. J. and Stinaff, R. D., "Minimum-Distance Bounds for Binary Linear Codes," IEEE Transactions on Information Theory, Vol. IT-19, pp. 344-356, May 1973.

[9] Forney, G. D., Concatenated Codes, M.I.T. press, Cambridge, Mass., 1966.